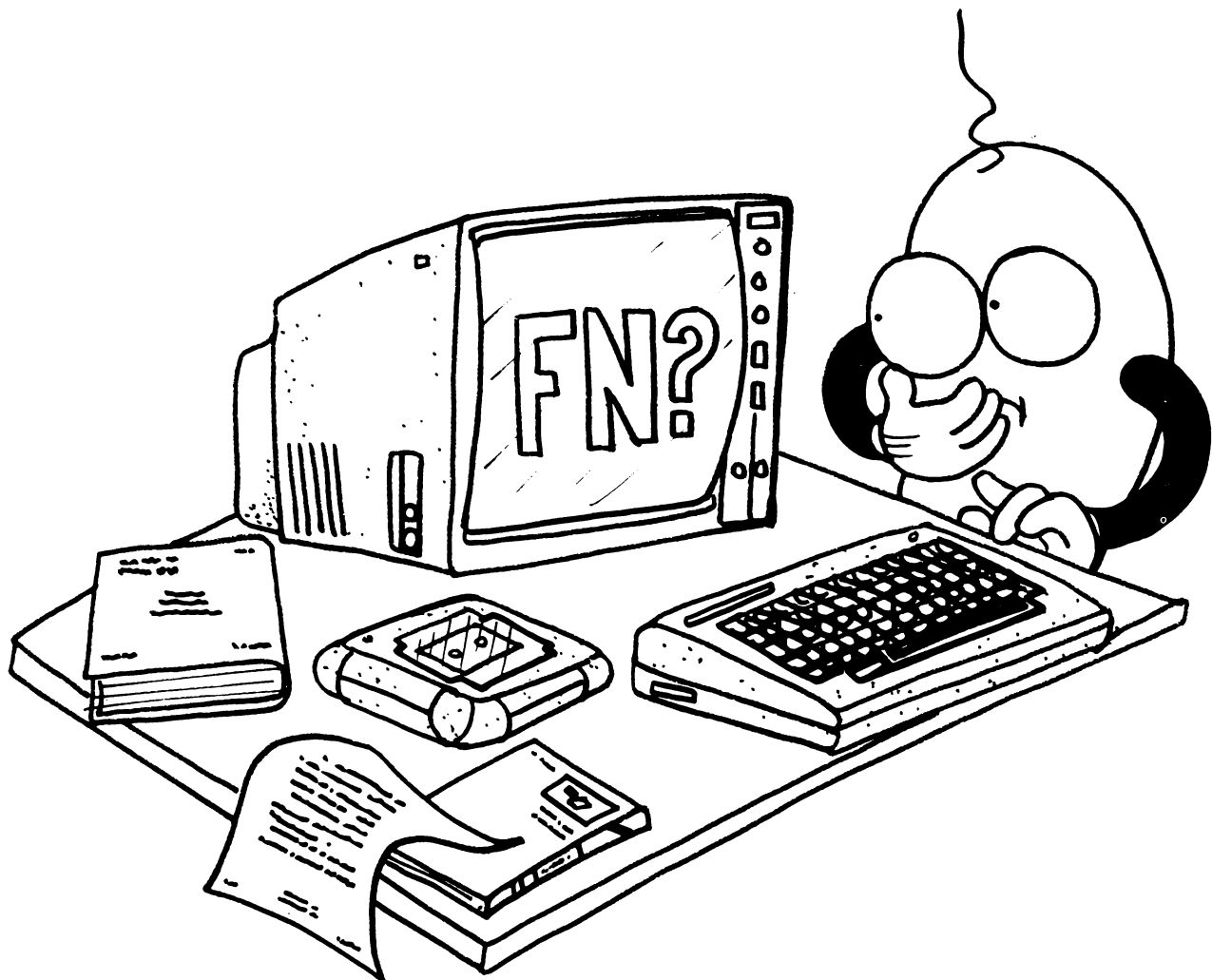
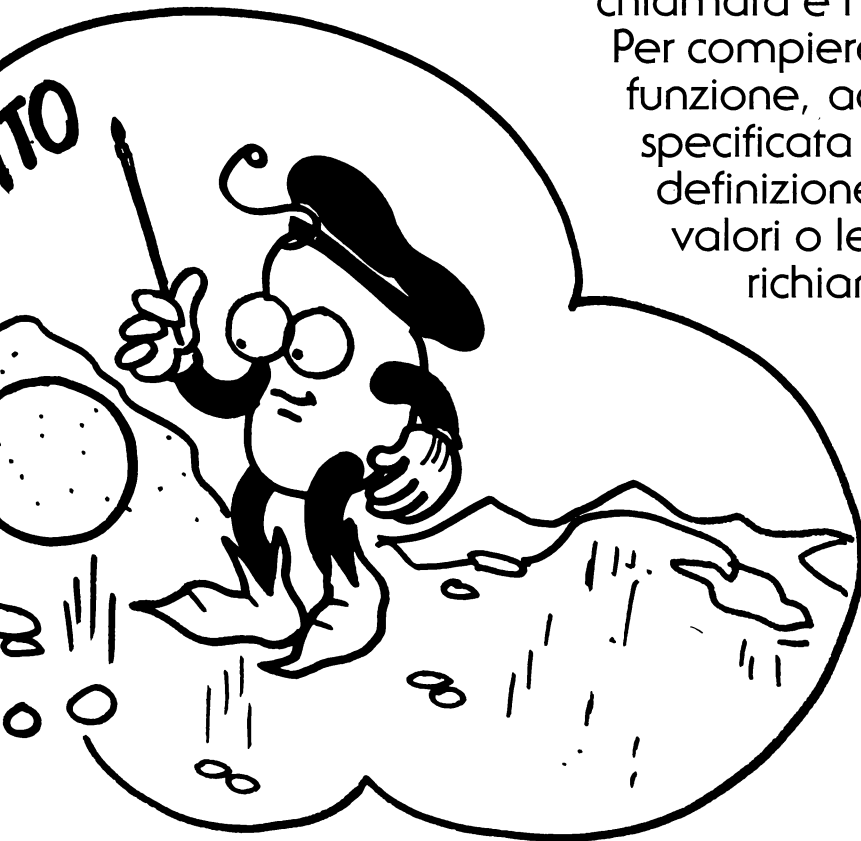
CONOSCERE IL COMPUTER DIRETTAMENTE DAL COMPUTER

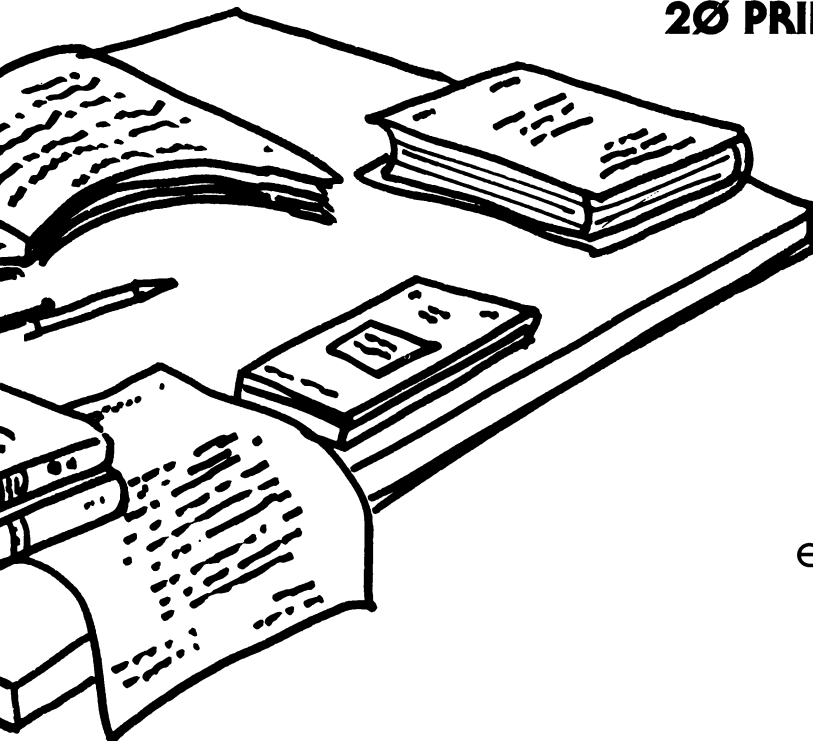
per Commodore Vic20 e 64

Abbiamo già visto le funzioni matematiche (lez. 12) e quelle stringa (lez. 15), ora ti spiegherò come definire nuove funzioni che, naturalmente, potranno contenere anche quelle già viste.
Questo è utile soprattutto per le applicazioni matematiche, quando è necessario usare la stessa funzione in parecchi punti del programma. L'istruzione che ti permette di definire una nuova funzione è DEF, usata nella forma:

DEF FNA (A,B) =A + A * B

Nel nome della funzione **FNA**, le prime due lettere devono sempre essere **FN**, la terza invece la dovrai scegliere tu.
(A,B) è una lista di argomenti, in questo caso sono 2.
A + A * B è una funzione che utilizza gli argomenti posti tra le parentesi.
La funzione può avere un numero maggiore di variabili, rispetto a quelle presenti come argomenti tra le parentesi.
Puoi definire sia funzioni numeriche che stringa.

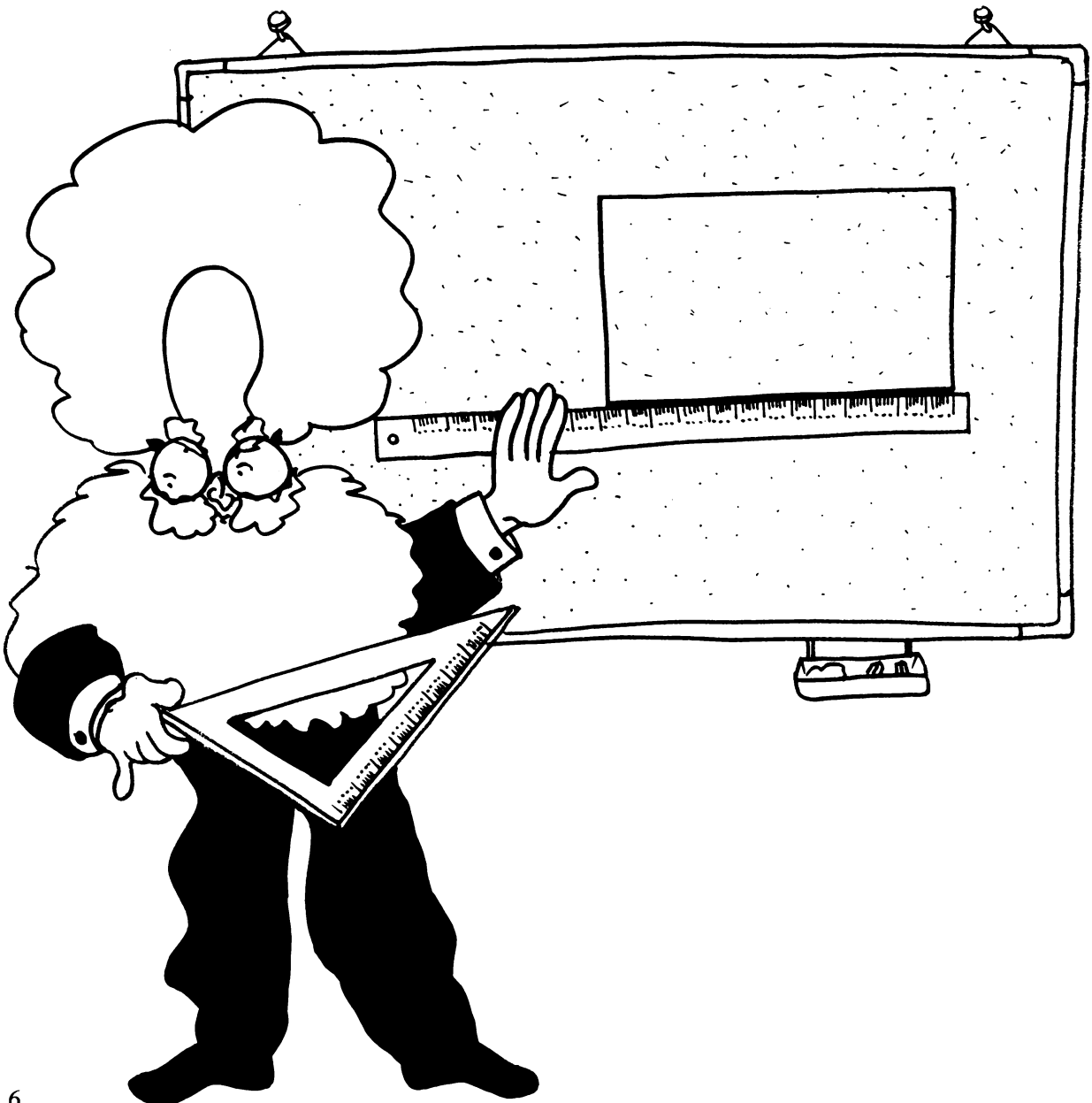
Ti consiglio di scrivere le definizioni all'inizio del programma, questo perchè sarà più facile poi ritrovarle. Per richiamare una funzione basta specificarne il nome, seguito da tanti argomenti tra le parentesi quanti sono quelli della definizione.
Non è necessario che i nomi degli argomenti della definizione siano uguali a quelli specificati nel richiamo. Puoi usare il richiamo di una funzione come se fosse una comune variabile e quindi metterla anche all'interno di espressioni.

Nel calcolo di una funzione i valori da assegnare agli argomenti vengono specificati dalla chiamata e non dalla definizione. Per compiere il calcolo della funzione, ad ogni variabile specificata come argomento della definizione, verranno assegnati i valori o le variabili usate nel richiamo.

Ad esempio in:

1Ø DEF FNA (R) = R ✳ R ✳ 3.14
2Ø PRINT FNA (6)

La linea 1Ø definisce la funzione che calcola l'area del cerchio. La 2Ø stampa il risultato della funzione richiamata usando l'argomento 6. Quindi il computer assegnerà R = 6 e calcolerà: **6 ✳ 6 ✳ 3.14**

Oppure:

```
10 DEF FNR (B,A) = B * A
20 K = 3
30 Z = FNR (K,6)
```

La linea 1Ø definisce la funzione che calcola l'area del rettangolo.
La 2Ø compie l'assegnazione K = 3.
La 3Ø assegna a Z il risultato della funzione richiamata usando gli argomenti K e 6.
Quindi il computer assegnerà B=K, cioè B=3 e A=6 calcolando: **3 * 6**.

Listato dell'esercizio: ESEMPIO PRATICO PER L'UTILIZZO DEL DEF

```
1Ø deffnv(t)=9.8*t
2Ø deffns(t)=(9.8*t↑2)/2
25 print"tempo velocità spazio"
26 print" sec   mt/sec    mt"
3Ø fork=1to1Ø
4Ø v=fnv(k):s=fns(k)
5Ø printk;tab(7);v;tab(15);s
7Ø nextk
```

Listato dell'esercizio: COMPLETA IL LISTATO

```
1Ø printchr$|*|
2Ø print
3Ø forx=1to|*|
4Ø r=|*|(rnd(Ø)*26+65)
5Ø a$(x)=|*|(r)
6Ø nextx
7Ø s=|*|:print"   ";
8Ø forx=1to1Ø
9Ø printa$(x);
1ØØ a$=|*|+a$(x)
11Ø next|*|:print
12Ø input|*|
13Ø f=ti
14Ø ift$=a$then|*|
15Ø print"hai sbagliato"
16Ø end
17Ø print"secondi:";(f-s)/|*|
```

PROGRAMMIAMO INSIEME (CBM 64)

```
10 poke53280,0:poke53281,9
20 printchr$(147)chr$(5)
30 for k=1 to 3
40 read s$(k),c$(k)
50 next k
60 for r=1 to 3
70 print"capitale della ";s$(r)
80 input"risposta";r$
90 ifr$=c$(r)thenprint"esatta":e=e+1:goto 110
100 print"sbagliata"
110 next r
120 print"risultati:"
130 print"domande n. 3"
140 print"risposte esatte n.";e
200 data"grecia","atene","francia","parigi","gran bretagna","londra"
```

PROGRAMMIAMO INSIEME (VIC 20)

```
10 poke36879,136
20 printchr$(147)chr$(5)
30 for k=1 to 3
40 read s$(k),c$(k)
50 next k
60 for r=1 to 3
70 print"capitale della ";s$(r)
80 input"risposta";r$
90 ifr$=c$(r)thenprint"esatta":e=e+1:goto 110
100 print"sbagliata"
110 next r
120 print"risultati:"
130 print"domande n. 3"
140 print"risposte esatte n.";e
200 data"grecia","atene","francia","parigi","gran bretagna","londra"
```

Soluzione dell'esercizio: COMPLETA IL LISTATO

(lez. 19)

```
10 DIMA$(15),B$(15)
20 INPUT"QUANTI NOMI(1-15)";N
30 IFN>15THEN20
40 FORK=1TON
50 INPUT"NOME";A$(K)
60 NEXTK
70 H=1
80 FORI=1TON
90 FORK=1TON
100 IFA$(I)<A$(K)THENH=H+1
110 NEXTK
120 IFB$(H)<>""THENH=H+1:GOTO120
130 B$(H)=A$(I):H=1:NEXTI
140 FORK=NTO1STEP-1
150 PRINTB$(K)
160 NEXTK
```